

**EKSAMEN**

**I**

**6108/6108N PROGRAMMERING I JAVA**

**20. 05. 2016**

<b>Tid:</b>	4 timer
<b>Målform:</b>	Bokmål / Nynorsk
<b>Sidetall:</b>	Framsida + 11 sider
<b>Hjelpemiddel:</b>	Alt trykt og skriftlig materiale.
<b>Merknader:</b>	Vekting er gitt ved starten av hver oppgave. Kandidaten kan som en del av besvarelsen bruke kall på metoder i Min-klassen (se vedlegg).
<b>Vedlegg:</b>	Min-klasse. Se side 11.

## BOKMÅL

## Oppgave 1

\*\*\* FLERVALG \*\*\*

Vekting: 20%

Angi spørsmålsnummer sammen med valgt svaralternativ, f.eks. slik: 1 a)

Flervalgsdelen består av 12 spørsmål, der hvert spørsmål har kun ett riktig svaralternativ. Det er anledning til å gardere (dvs. angi flere svaralternativ). Riktig svaralternativ gir 3 poeng, (hvert) feil svaralternativ gir -1 poeng, mens et ubesvart spørsmål gir 0 poeng.

**NB!** Du kan forutsette at nødvendige import-setninger er på plass i alle kodeutsnittene.

## SPØRSMÅL:

1. Hva er verdien i variabelen w etter denne tilordningen?

- a) 1
- b) 2.25
- c) 5
- d) Ingen verdi pga. syntaksfeil

```
int u = 5;
int w = u/2*2+1;
```

2. Hva er verdien i variabelen s etter denne tilordningen?

- a) 0
- b) 4.5
- c) 8
- d) Ingen verdi pga. syntaksfeil

```
int t = 9;
int s = 9 - t%2;
```

3. Hvilket uttrykk gir samme effekt på variabelen x som denne tilordningen?

- a) x++;
- b) ++x;
- c) x+=2;
- d) Umulig å si pga. syntaksfeil

```
int x = x + 2;
```

4. Hva blir utskriften i konsollet dersom innholdet i int-variabelen c er lik 1?

- a) v = 1 + 1
- b) v = c + 1
- c) v = 11
- d) Ingen utskrift pga. syntaksfeil

```
out.println "v = " + c + " + 1";
```

5. Hvis verdien i variabelen x er 1 umiddelbart etter at if-setningen er avsluttet, hva var da verdien i variabelen y?

- a) -1
- b) 0
- c) 1
- d) Umulig å si pga. syntaksfeil

```
int x = y; // y har int-verdi
if ( x < 0 || x > 0 )
    x = -1;
else
    x = 1;
```

6. Hva er verdien i variabelen v umiddelbart etter at **for**-løkka er avsluttet?

- a) 0.0
- b) 1.0
- c) 3.0
- d) Ingen verdi pga. syntaksfeil

```
double v = 0.0;
for (int i=1; i<=5; i++)
    v += i/5;
```

7. Hva blir skrevet i konsollet etter denne sekvensen av programsetninger?

```
String t = "Antall av liten a";
char b = 'a';
for (int i=0; i<t.length(); i++)
    if ( t.charAt(i) == b )
        out.print( (i+1) + " " );
```

- a) Antall forekomster av bokstaven liten a i tegnsekvensen t
- b) Alle forekomster av bokstaven liten b i tegnsekvensen t
- c) En serie nummer som viser til plasser der bokstaven liten a står i tegnsekvensen t
- d) En feilmelding pga. syntaksfeil

8. Hva blir skrevet i konsollet etter denne sekvensen av programsetninger?

- a) 4 = 2\*2
- b) 3 = 2\*2
- c) true
- d) En feilmelding pga. syntaksfeil

```
int[] i = {4, 3, 2, 1, 0};
i[2] = i[1] - i[3];
i[3] = i[4] + i[0];
out.println(i[3] + " = 2*" + i[2]);
```

9. Hva blir skrevet i konsollet etter denne sekvensen av programsetninger?

- a) sum = 199
- b) sum = BAD
- c) sum = c1c0c3
- d) En feilmelding pga. syntaksfeil

```
char[] c = {'A', 'B', 'C', 'D'};
// A har tallkode 65, B 66, osv.
out.println("sum = " + c[1] + c[0] + c[3]);
```

10. Hva blir returverdien fra metodekallet met('A')? Hjelpemetoden er gitt ved:

- a) 'a'
- b) 97
- c) -1
- d) Ingen verdi pga. syntaksfeil

```
private static int met(char t) {
    if ('A' < t && t < 'Z')
        return t + 32;
    return -1;
}
```

11. Hvor mange ganger blir setningen i denne løkke-kroppen utført?

- a) 5
- b) 8
- c) Ukjent antall pga. evig løkke
- d) Ingen utførelser pga. syntaksfeil

```
int w = 16;
do { w = w/2; } while (w > 0);
```

12. Hvilket alternativ vil du si omtaler *datatyper* mest korrekt?

- a) I en tabelldeklarasjon er det datatypen som angir antall komponenter
- b) Klasser med `toString`-metoder har alltid datatype på konstruktøren(-e)
- c) Datatyper brukes til å vise hva som er forventet returverdi fra `void`-metoder
- d) Datatypen avgjør bl.a. hvor og hvordan en verdi blir lagret i hurtigminnet

\*\*\* SLUTT PÅ FLERVALG \*\*\*

## Oppgave 2

Vekting: 15%

- a) I tekststrengen `dato` er det en dato på formatet "dd.mm.åååå", der det først ligger et dagnummer (2 siffer), så et månedsnummer (2 siffer) og til slutt et årstall (4 siffer). Skriv kun de Java-instruksjonene (*ikke* et komplett program) som er nødvendige for å få den samme datoen i tekststrengen `snuDato`, men da i et omsnudd format "ååååmmdd" (der d, m og å har samme tolkning som ovenfor) uten punktum som skille tegn.

F.eks. `dato = "20.05.2016"`       $\longrightarrow$       `snuDato = "20160520"`

- b) Vi trenger en Java-metode som kan avgjøre hvor mange siffer det er i et heltall. Denne metoden skal inngå i `Min`-klassen (jfr. vedlegget). Skriv en slik Java-metode som tar en heltallsparameter, finner antall siffer i dette heltallet og returnerer antall siffer. F.eks. om vi sender inn heltallet 24680 så skal metoden returnere 5.
- c) Skriv kun de Java-instruksjonene som er nødvendige (*ikke* et komplett program) for å ta et tallsett som allerede ligger i (og fyller hele) desimaltallstabellen `desTall`, og splitte dette tallsettet i to: Alle de positive tallene *kopieres* til tabellen `posTall`, mens alle de negative tallene og evt. nuller ikke blir berørt.

Den nye tabellen skal ikke ha noen tomme posisjoner, dvs. at først må det avgjøres hvor mange positive tall det er i tallsettet før oppretting av denne nye tabellen.

**Oppgave 3****Vekting: 25%**

På sekvensielle tekstfiler ligger det informasjon om innholdet i spillelister for en musikk-tjeneste. Det er én spilleliste pr. fil, og hver fil har tre felt pr. linje: Spornummer, sangtittel og spilletid. Alle spilletider er angitt ved oppsettet «minutt:sekund».

Feltene er adskilt med minst ett blankt tegn. F.eks. kan spillelista som ligger på fila *kygo.txt* se slik ut:

kygo.txt		
01	Stay	3 : 59
02	Stole the Show	3 : 42
03	Firestone	4 : 33
04	Fragile	3 : 50
05	Raging	3 : 44

Det er ikke beregnet total spilletid i noen av filene, så det blir da utfordringen her. Skriv et komplett Java-program, som etter å ha lest inn filnavnet fra brukeren, går gjennom fila og beregner total spilletid for den tilhørende spillelista.

Programmet skal skrive beregnet spilletid inn igjen på den samme fila som en ekstra tekstlinje helt til slutt, f.eks. slik: "Total spilletid 19:48".

**Oppgave 4****Vekting: 10%**

For å holde orden på vinsamlingen sin, bruker et hotell en Java-løsning basert på klassen **Vin**. Objekt fra **Vin**-klassen har følgende informasjon om en gitt vinflaske: Farge (R=rød/H=hvit), navn, produksjonsland, produksjonsår og eventuelt druetype (ikke aktuell for alle vinflasker).

I **Vin**-klassen brukes følgende navn på objektvariablene: *farge*, *navn*, *land*, *år* og *drue*. Alle disse variablene har «private access», dvs. klassen benytter teknikken med innkapsling. **Vin**-klassen skal *ikke* programmeres.

I *vinkart* (en 1-dimensjonal tabell med referanse til **Vin**-objekt) er det informasjon om noen aktuelle viner for en større middag på hotellet. I tabellen er vinene allerede sortert etter årstall, fra eldste til yngste, men for øvrig er ikke vinene plassert i noen slags ordning.

Skriv *kun* de Java-setningene som er nødvendige for å få skrevet ut hele dette vinkartet (all informasjon som er registrert), der alle de røde vinene kommer først sortert etter årstall og så alle de hvite vinene sortert etter årstall (i begge tilfeller fra eldst til yngst). Skriv til konsollet og pass på å bruke tydelige overskrifter for å skille mellom de to vinlistene.

**Oppgave 5****Vekting: 30%**

En enkel koding av korte meldinger foregår ved at hvert opprinnelig par av bokstaver bytter plass. Dvs. bokstav 1 og 2 bytter plass, 3 og 4 bytter plass, osv. Om ikke meldingen har et partall antall bokstaver, så føyer vi bare til et mellomrom til slutt i meldingen.

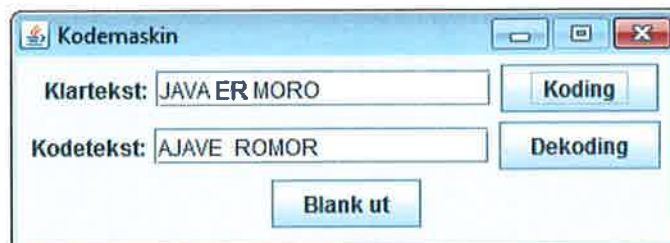
For eksempel:

M E L D I N G    →    E M D L N I G

J A V A    E R    M O R O    →    A J A V E    R O M O R

Skriv et komplett Java-program, som basert på et grafisk brukergrensesnitt (GUI), både kan kode fra en klartekst (lesbar melding) til en kodetekst, og dekode fra en kodetekst til en klartekst. Det skal kun være korte meldinger, så det holder med tekstfelt (JTextField) til klartekst og kodetekst. Det bør være en knapp for å kode, og en for å dekode. Programmet bør også ha en knapp for å blanke ut tekstfeltene før evt. neste omkoding.

Legg vekt på å få et GUI som ligner på det som er gitt under, der begge de to tekstfeltene har lengde 16.



130 pix

360 pix

— SLUTT —

NYNORSK

**Oppg ve 1**

\* \* \* FLEIRVAL \* \* \*

**Vekting: 20%**

Oppgi sp rsm lsnummer saman med valt svaralternativ, f.eks. slik: **1 a)**

Fleirvalsdelen består av 12 sp rsm l, der kvart sp rsm l har kun eitt rett svaralternativ. Det er lov   gardere (dvs. oppgi fleire svaralternativ). Rett svaralternativ gir 3 poeng, (kvart) feil svaralternativ gir -1 poeng, medan eit ubesvara sp rsm l gir 0 poeng.

**NB!** Du kan forutsette at naudsynte import-setningar er p  plass i alle kodeutsnitta.

### SP RSM L:

1. Kva er verdien i variabelen *w* etter denne tilordninga?

- a) 1
- b) 2.25
- c) 5
- d) Ingen verdi pga. syntaksfeil

```
int u = 5;
int w = u/2*2+1;
```

2. Kva er verdien i variabelen *s* etter denne tilordninga?

- a) 0
- b) 4.5
- c) 8
- d) Ingen verdi pga. syntaksfeil

```
int t = 9;
int s = 9 - t%2;
```

3. Kva for eit uttrykk gir samme effekt p  variabelen *x* som denne tilordninga?

- a) `x++;`
- b) `++x;`
- c) `x+=2;`
- d) Umogleg   seie pga. syntaksfeil

```
int x = x + 2;
```

4. Kva blir utskrifta i konsollet dersom innhaldet i *int*-variabelen *c* er lik 1?

- a) `v = 1 + 1`
- b) `v = c + 1`
- c) `v = 11`
- d) Inga utskrift pga. syntaksfeil

```
out.println "v = " + c + " + 1";
```

5. Om verdien i variabelen *x* er 1 umiddelbart etter at *if*-setninga er avslutta, kva var d  verdien i variabelen *y*?

- a) -1
- b) 0
- c) 1
- d) Umogleg   seie pga. syntaksfeil

```
int x = y; // y har int-verdi
if ( x < 0 || x > 0 )
    x = -1;
else
    x = 1;
```

6. Kva er verdien i variabelen v umiddelbart etter at **for**-løkka er avslutta?

- a) 0.0
- b) 1.0
- c) 3.0
- d) Ingen verdi pga. syntaksfeil

```
double v = 0.0;
for (int i=1; i<=5; i++)
    v += i/5;
```

7. Kva blir skrevet i konsollet etter denne sekvensen av programsetningar?

```
String t = "Antal av liten a";
char b = 'a';
for (int i=0; i<t.length(); i++)
    if ( t.charAt(i) == b )
        out.print( (i+1) + " " );
```

- a) Antal forekomstar av bokstaven liten a i teiknsekvensen t
- b) Alle forekomstar av bokstaven liten b i teiknsekvensen t
- c) Ein serie nummer som viser til plassar der bokstaven liten a står i teiknsekvensen t
- d) Ei feilmelding pga. syntaksfeil

8. Kva blir skrevet i konsollet etter denne sekvensen av programsetningar?

- a) 4 = 2\*2
- b) 3 = 2\*2
- c) true
- d) Ei feilmelding pga. syntaksfeil

```
int[] i = {4, 3, 2, 1, 0};
i[2] = i[1] - i[3];
i[3] = i[4] + i[0];
out.println(i[3] + " = 2*" + i[2]);
```

9. Kva blir skrevet i konsollet etter denne sekvensen av programsetningar?

- a) sum = 199
- b) sum = BAD
- c) sum = c1c0c3
- d) Ei feilmelding pga. syntaksfeil

```
char[] c = {'A', 'B', 'C', 'D'};
// A har tallkode 65, B 66, osv.
out.println("sum = " + c[1] + c[0] + c[3]);
```

10. Kva blir returverdien frå metodekallet met('A')? Hjelpemetoden er gitt ved:

- a) 'a'
- b) 97
- c) -1
- d) Ingen verdi pga. syntaksfeil

```
private static int met(char t) {
    if ('A' < t && t < 'Z')
        return t + 32;
    return -1;
}
```



11. Kor mange gonger blir setninga i denne løkke-kroppen utført?

- a) 5
- b) 8
- c) Ukjend antal pga. evig løkke
- d) Ingen utføringar pga. syntaksfeil

```
int w = 16;
do { w = w/2; } while (w > 0);
```

12. Kva for alternativ vil du seie omtalar *datatypar* mest korrekt?

- a) I ein tabelldeklarasjon er det datatypen som avgjer talet på komponentar
- b) Klassar med toString-metodar har alltid datatype på konstruktøren(-ane)
- c) Datatypar brukast til å vise kva som er forventa returverdi frå void-metodar
- d) Datatypen avgjer bl.a. kor og korleis ein verdi blir lagra i hurtigminnet

\*\*\* SLUTT PÅ FLEIRVAL \*\*\*

## Oppgåve 2

Vekting: 15%

- a) I tekststrengen `dato` er det ein dato på formatet "dd.mm.åååå", der det først ligg eit dagnummer (2 siffer), så eit månadsnummer (2 siffer) og til slutt eit årstal (4 siffer). Skriv kun dei Java-instruksjonane (*ikkje* eit komplett program) som er naudsynt for å få den same datoen i tekststrengen `snuDato`, men då i eit omsnudd format "ååååmmdd" (der d, m og å har same tolkinga som ovanfor) utan punktum som skiljeteikn.
- F.eks.    `dato = "20.05.2016"`     $\longrightarrow$     `snuDato = "20160520"`
- b) Vi treng ein Java-metode som kan avgjere kor mange siffer det er i eit heiltal. Denne metoden skal inngå i `Min`-klassen (jfr. vedlegget). Skriv ein slik Java-metode som tek ein heiltalsparameter, finn antal siffer i dette heiltalet og returnerer antal siffer. F.eks. om vi sender inn heiltalet 24680 så skal metoden returnere 5.
- c) Skriv kun dei Java-instruksjonane som er naudsynte (*ikkje* eit komplett program) for å ta eit talsett som allereie ligg i (og fyller heile) desimaltalstabellen `desTa11`, og splitte dette talsettet i to: Alle dei positive tala *kopierast* til tabellen `posTa11`, medan alle dei negative tala og evt. nullar ikkje blir rørt.
- Den nye tabellen skal ikkje ha nokon tomme posisjonar, dvs. at først må det avgjerast kor mange positive tal det er i talsettet før oppretting av denne nye tabellen.

**Oppgave 3****Vekting: 25%**

På sekvensielle tekstfiler ligg det informasjon om innhaldet i spelelister for ei musikk-teneste. Det er ei speleliste pr. fil, og kvar fil har tre felt pr. line: Spornummer, songtittel og speletid. Alle speletider er gitt ved oppsettet «minutt:sekund».

Felta er skilt frå kvarandre med minst eitt blankt teikn. F.eks. kan spelelista som ligg på fila *kygo.txt* sjå slik ut:

kygo.txt		
01	Stay	3:59
02	Stole the Show	3:42
03	Firestone	4:33
04	Fragile	3:50
05	Raging	3:44

Det er ikkje berekna total speletid i nokon av filene, så det blir då utfordringa her. Skriv eit komplett Java-program, som etter å ha lest inn filnavnet frå brukaren, går gjennom fila og bereknar total speletid for den tilhøyrande spelelista.

Programmet skal skrive berekna speletid inn igjen på den same fila som ei ekstra tekstline heilt til slutt, f.eks. slik: "Total speletid 19:48".

**Oppgave 4****Vekting: 10%**

For å holde orden på vinsamlinga si, brukar eit hotell ei Java-løysing basert på klassen **Vin**. Objekt frå **Vin**-klassen har følgjande informasjon om ei gitt vinflaske: Farge (R=raud/K=kvit), namn, produksjonsland, produksjonsår og eventuelt druetype (ikkje aktuell for alle vinflasker).

I **Vin**-klassen brukast følgjande namn på objektvariablane: *farge*, *namn*, *land*, *år* og *drue*. Alle desse variablane har «private access», dvs. klassen nyttar teknikken med innkapsling. **Vin**-klassen skal *ikkje* programmerast.

I **vinkart** (ein 1-dimensjonal tabell med referansar til **Vin**-objekt) er det informasjon om nokre aktuelle viner for ein større middag på hotellet. I tabellen er vinane allereie sortert etter årstal, frå eldst til yngst, men for øvrig er ikkje vinane plasserte i noko slags ordning.

Skriv *kun* dei Java-setningane som er naudsynte for å få skrevet ut heile dette vinkartet (all informasjon som er registrert), der alle dei raude vinane kjem først sortert etter årstal og så alle dei kvite vinane sortert etter årstal (i begge tilfelle frå eldst til yngst). Skriv til konsollet og pass på å bruke tydelege overskrifter for å skilje mellom dei to vinlistene.

**Oppgave 5****Vekting: 30%**

Ei enkel koding av korte meldingar skjer ved at kvart opprinnelege par av bokstavar byter plass. Dvs. bokstav 1 og 2 byter plass, 3 og 4 byter plass, osv. Om ikke meldinga har eit partal antal bokstavar, så føyer vi berre til eit mellomrom til slutt i meldinga.

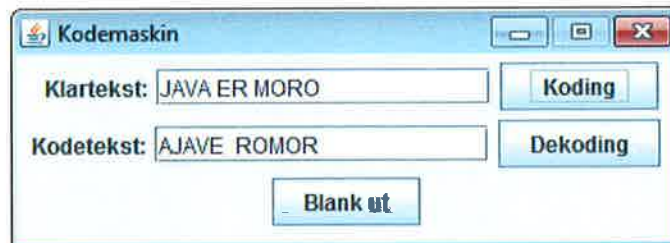
For eksempel:

M E L D I N G → E M D L N I G

J A V A E R M O R O → A J A V E R O M O R

Skriv eit komplett Java-program, som basert på eit grafisk brukargrensesnitt (GUI), både kan kode frå ein klartekst (lesbar melding) til ein kodetekst, og dekode frå ein kodetekst til ein klartekst. Det skal kun vere korte meldingar, så det held med tekstfelt (JTextField) til klartekst og kodetekst. Det bør være ein knapp for å kode, og ein for å dekode. Programmet bør og ha ein knapp for å blanke ut tekstfelta før evt. neste omkoding.

Legg vekt på å få eit GUI som liknar på det som er gitt under, der begge dei to tekstfelta har lengde 16.



130 pix

360 pix

— SLUTT —

## VEDLEGG

```

/*
*****
* Min: Verktøyklasse - dvs. samling av klassemetoder. Det er lov å
* kalle på disse metodene fra program der det måtte være aktuelt.
*****
*/
import static javax.swing.JOptionPane.*;
import static java.lang.Integer.*;
import static java.lang.System.*;
import static java.lang.Math.*;

public class Min {

    // Metoden lager en sekvens av tegn som en String
    public static String sekvens(char t, int antall) {
        String ut="";
        for (int i=1; i<=antall; i++)
            ut += t;
        return ut;
    }

    // Metoden avgjør om et tegn er en (engelsk) bokstav
    public static boolean erBokstav(char tegn) {
        return ('A' <= tegn && tegn <='Z') || ('a' <= tegn && tegn <='z');
    }

    // Metoden avrunder et tall til én desimal
    public static double avrund1(double tall) {
        return (int)(tall*10 + 0.5)/10.0;
    }

    // Metoden avrunder et tall til to desimaler
    public static double avrund2(double tall) {
        return (int)(tall*100 + 0.5)/100.0;
    }

    // Metoden leser inn et heltall i området (min-max)
    public static int lesHeltall(int min, int max) {
        int antall=0;
        do {
            String innTekst = showInputDialog("Gi heltall (" + min + "-" + max + "): ");
            antall = parseInt(innTekst);
            if ( antall < min || antall > max )
                showMessageDialog(null, "Ulovlig verdi!");
        } while (antall < min || antall > max);
        return antall;
    }

    // Metoden trekker et tilfeldig heltall i området min - max
    public static int trekkTall(int min, int max) {
        return min + (int)( random()*(max-min+1) );
    }

    // Her kommer det en metode som teller antall siffer i et heltall
}

```